Science-based System Architecture Drivers for the ECS Project

(A White paper)

Revision 1.0

December 7, 1993

Hughes Applied Information Systems, Inc.

1616A McCormick Dr.
Landover, MD 20785
(301) 925-0300

193-00611

Contents

1. Introduction

1.1. Purpose

In this draft of the Science-based System Architecture Drivers document, we attempt to capture and summarize key architectural issues emanating from SRR, and from post-SRR discussions with members of the scientific community. These drivers are being summarized and returned to the science community for validation — to make sure that we have appropriately captured the needs expressed by representatives of that community, and to attempt to reach a more broad consensus among prospective ECS users as to fundamental system architecture drivers.

The drivers captured here are intended to reflect significant issues raised by the science community in reviewing the baseline architecture; there are additional system architecture drivers and constraints that are not covered in this document. It is anticipated that the "science drivers" discussed in this document will be combined with additional drivers to push out a revised set of system requirements in preparation for the System Design Review (SDR). The document in which the entire set of system drivers is pulled together has yet to be defined.

The drivers discussed here, along with the EOSDIS Advisory Panel's higher level recommendations to the ECS project, an SRR RID review, Version 0 analysis, initial external system analysis, and an analysis of the potential for other uses of ECS, will form the foundation of the conceptual architecture framework for the December review. These sources will be used to develop a high level architectural "mandate," key characteristics that the ECS architecture must exhibit to meet the needs of the science community at large. From this mandate, a conceptual architecture will be developed, which provides the framework within which the ECS system design will continue. The conceptual architecture forms the foundation for key architectural "trades," many of which have begun in preparation for the System Design Review. A number of these will be reported on during the December review.

Finally, feedback from SRR and the post-SRR visits, as well as the EOSDIS Advisory Panel's recommendations to the ECS project, have been used to develop new or augmented processes and relationships. These new processes and relationships are briefly presented.

1.2. Approach/Format

This paper is derived largely from the feedback received during visits with the following members of the scientific community:

- Colorado University, Boulder (Dr. Bill Emery, et al)
- University of New Hampshire (Dr. Berrien Moore, et al)
- Oregon State University (Dr. Mark Abbott, et al)
- University of California, Santa Barbara (Dr. Jeff Dozier, et al)

As indicated by the list, attempts have been made to visit in particular with some of the interdisciplinary investigation teams, as these teams have been most vocal in expressing their concern over materials presented at SRR. Additionally, it was our goal to better understand the facilities and operations of some of these research environments, in order to better understand their system perspective. Because these visits are ongoing, this document represents a snapshot of our understanding. As additional visits are completed, our understanding will be augmented to incorporate new needs.

From these visits and discussions, we have attempted to abstract some of the fundamental architectural "drivers," as we see them. We have summarized these drivers in the following sections, attributing significant concepts to their source(s) either by name or by institution (where several individuals at the institution contributed to the collective concept). Where concepts were mentioned by most or all of the interviewed sources, we have identified such items with the parenthetical note "(consensus)." Where these comments were derived from discussions at SRR, this fact is denoted without a more specific reference. A number of the investigators' comments could be applied to more than one driver. Rather than repeating investigators' comments, we have attempted either to categorize comments with the primary corresponding driver, or to break comments up across multiple drivers as appropriate.

1.3. Parallel activities not covered

This paper does not directly address the results of parallel activities that are also being worked into the December review (V0 analysis, analysis of "external" systems, RIDs). Rather, we have attempted to capture the key drivers as voiced by the science "users."

1.4. Remainder of document

The remainder of this document is organized as follows:

- Section 2 outlines key architectural "drivers" that we extracted from our discussions with the science teams. For each of the drivers identified, a description of our understanding and appropriate references is provided.
- Section 3 attempts to collect the drivers into a high level architectural "mandate," identifying key architectural features that the ECS needs to move toward.
- Finally, section 4 touches on some of the changes in relationships with the science community and in modifications to processes where greater science community involvement is desired.

2. Architectural Drivers

2.1. Introduction

This section outlines some of the architectural drivers we were able to extract from our series of interviews. We have attempted to summarize (in our own words) each of the key drivers, in order to test our understanding of the concepts discussed, and to attempt to extract the key common issues from multiple related but slightly varying concepts. These summaries are followed, where appropriate, by extractions from trip reports summarizing key discussion topics. These topics are attributed to the team(s) or individual(s) who presented those concepts. Concepts that were presented by more than one team are indicated by the parenthetical note "(consensus)."

In the following two subsections, we present the drivers organized as follows:

Science (user) drivers	Issues associated with scientist's direct interaction with the system, or with concerns about how the system might limit or promote future scientific analysis.
Technology drivers	Technology advances that need to be accommodated

The order of the drivers within these categories is somewhat random, though attempts have been made to put what we believe to be the strongest messages at the top of each of the two lists.

2.2. Science (user) drivers

This subsection focuses on drivers that directly affect scientists' use of the system, or that might limit or promote future use of the system in support of scientific analysis.

1. Facilitate an efficient data search and "access" paradigm

The "search and order" paradigm is potentially too heavyweight and too beauraucratic. A lighter weight "search and access" paradigm should be employed, in which, once objects have been identified through specification in a search operation, they can simply be accessed (i.e., passed to an application, "opened," etc.).

If we draw an analogy to a Unix file system, users "search" for data objects by traversing an information-rich hierarchical file system. The pathname and file extension represent "metadata" about the desired object, organized into a hierarchical namespace. The user "refines" his query (by repeatedly *cd*'ing down the directory structure) to the point where he has located the desired object. He then has a "context" in which to apply various operations to that object (i.e., invoke applications on it).

In Unix, files are represented in the inode structures, with hierarchical names "attached" for user readability. Similarly, in ECS, objects will have some object id and some set of parameters

(metadata) that are useful in finding that object. Whether an actual hierarchical file system exists or not is immaterial. The key is that, once found, an object ought to be able to be operated on directly (i.e., without the need to "place an order" for the data).

The Andrew File System and OSF DFS (Distributed File System) extend the Unix file system concept into a global namespace, with provisions for data distribution, replication, and migration to provide better data availability and performance in large distributed configurations. Similarly, ECS should support a global object space made up of physically distributed components that can be managed in a similar fashion to support the data "access" needs of the science community.

Substantiating comments

(Consensus) Concern was voiced at SRR regarding the centralized, "heavyweight" nature of the product "ordering" scenario. Belief in the interdisciplinary science community is that data "access," as opposed to "ordering" is the logical end to the search process. It was suggested that the Andrew File System might be a model for how to provide users with "access" to a global object space once the desired object has been identified.

2. Support a dynamic product lifecycle and easily extensible product set

The system needs to support a product lifecycle that is more dynamic than the one accommodated by the baseline design. Data "products" are in actuality a mixture of analysis and modeling (pixel "mixing" models — e.g., atmospheric correction models, dispersion models, etc.). Even gridding and binning schemes may use science-specific models that can differ across scientific domains. Scientists are continually refining these models to provide more accurate products. These products will vie for "shelf space" in the ECS data "supermarket," with newer products replacing older ones on an ongoing basis once an "acceptance threshold" is exceeded. Figure 1 illustrates both a step model, and a more accurate consensus-based model of a product's use through time. The latter scenario requires a streamlined publishing and distribution process to allow potential products to get "air time" and thus be considered for wider availability.

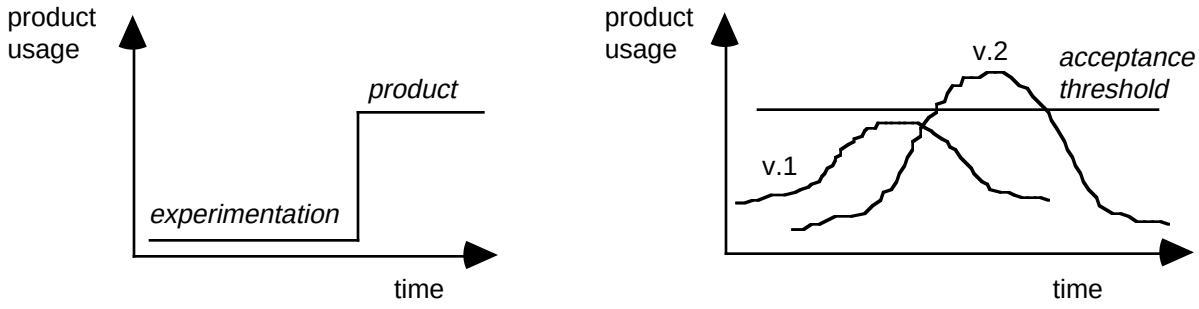

Figure 1. Dynamic product lifecycle characteristics

Because of this dynamic lifecycle, there will be less demand for longer term "static" products, and more demand for source data (e.g., level 1b) and the computational resources to derive suitable level 2 and 3 products on-the-fly. This has more of an object-oriented flavor, where data objects are instantiated at the time of request, using appropriate source data objects (level 1b) and methods (which may even have been specified by the user). This approach will also support evolution of the data repository, as new methods can be developed to "instantiate" new data products. This capability for growth is key to ensuring the long term viability and utility of the ECS data archive.

Because algorithms are being continually developed and refined at the SCFs, it may make sense to support SCFs as data providers (either of the data, or of the "methods") for short periods of time, before an algorithm is "accepted" by the community at large. Data in this state should not be withheld from the system simply because of its "experimental" nature. The ability to support experimental algorithms and data products further drives the need for the system to maintain appropriate data "heritage" information (see science driver 5, below). Once an algorithm has gained acceptance, it could be migrated to an appropriate DAAC (using suitable algorithm integration and test procedures) for long term maintenance.

Additionally, supporting SCFs as data providers allows for the application of additional resources (e.g., SCF processing capacity) to the task at hand. This could be helpful in supporting aggressive "process on demand" scenarios, content-based search, and the application of user specified methods.

Substantiating comments

(Emery) The need for "product" sources other than the DAACs (i.e., the SCFs) is motivated by the fact that algorithm consensus "happens" (by people using the data products), and can not be accomplished through peer review. This belief is the driving force behind the need for a more distributed "production" capability than is supported by the current architecture and system concept. Bill believes that the predominant requests in the scientific research community will be for level 1B data, not higher level products. Researchers should be able to provide ("publish") new products to the system. New products that generate a lot of interest should then be "migrated" to the DAACs for routine production.

(Moore) Prof. Moore pointed out that sometimes the SCFs might be the best place to produce data products since they have more computing power than many DAACs and they are the source of the algorithms that transform sensor data into products. He did admit that when a particular process must be done year-after-year without change, that the SCFs are not the right place. This brought up the idea that data product creation might move over the life of ECS , from the SCF to the DAAC, back to the SCF when an algorithm is fixed or improved, and then back to the DAAC.

(Vorosmarty/Aber - UNH) ... discussed the difficulties in obtaining and processing large amounts of data. They used different processing algorithms (e.g.,

multispectral) over diverse data formats. The algorithms are not standardized yet, and are still in the experimental stage. That's why they don't think that it is feasible to collect and store all the algorithms in one place. Other issues such as getting the data in a timely and user-friendly fashion is a big concern to the scientific community.

(OSU) Concerning data formats, in general they prefer translators over standard formats. Abbott claims to subscribe to Stonebraker's philosophy of "all you need is the recipe." That opinion was loudly seconded by John Gregor. In addition Gregor feels that there are relatively few operators that are required for content-based queries. In addition, he feels (as do all the OSU people) that it should all be computed on the fly (perhaps with caching).

(Richman - OSU) There is often a lack of unanimity in the science community about the data processing algorithm or the application of that algorithm to a specific problem. It is unacceptable for the processing center to offer the product to the researcher in the processed form (after application of a particular algorithm selected by the ECS). There are real uncertainties in all ocean models that require tradeoffs between use of data and theory — he used an example of the effect of ice thickness on subsurface thermal characteristics — cannot be obtained from either theory or measurement alone; it requires both, and the ECS must accommodate this need.

(Abbott) if you're a global change researcher, you'll want to pull in data from everywhere on sea surface temperature. Existing data retrieval algorithms or canned products won't generally be applicable. For his work, Abbott stresses the need for the availability of raw (unprocessed) data because the generation of quality high level products changes dramatically with time. A scientist will want to apply his own tools, but when he uses complementary data (e.g., ozone field distributions) he'll want to be able to pull in the "latest and greatest". This means communicating with a high power, yet flexible system.

(Abbott) There should be a few core products that one can always find in a predictable location, and other than that, products vie for shelf space depending on some form of "economic value". Abbott's idea of economic value is some measure of use by the scientific community.

(OSU) Consistent with other viewpoints about the value of L2 and L3 data for scientific research, they site a paper by Ralph Kahn at JPL which shows error magnitudes of the same order as the signal when gridding to one resolution and sampling at another. This has been cited by a number of researchers as a reason why a great deal of processing will really occur at the SCFs, and why scientifically interesting "products" will really be SCF produced, and not DAAC produced.

(OSU) Standard products will not address the needs of Global Change research. This research requires a blend of numerous data sources that must be processed in similar ways. Hence, reversion to lower level "products" (L1) is required to ensure consistency in modeling, gridding, binning, etc. For example, radar altimeter data is only as good as the full set of corrections used (orbital mechanics, atmospheric corrections, etc.). Understanding of these "models" is changing all the time, requiring regeneration of products as a routine operating procedure. Hence, data "products" are really algorithmic representations that are instantiated at request time.

(Walstad - OSU) In the past, ocean modeling was based on the assumption that ocean currents and eddies can't be resolved. The computing power now available makes local scale processes more accessible. This will greatly change the applicability of mesoscale models. The data system must be sufficiently versatile to accommodate these kinds of changes.

(OSU) The nature of queries will change over time. The ECS system must be flexible enough to handle changing types of queries. The use of time and location is not sufficient — there must be support for subsetting within the parameter space (e.g., show me the data with temperature gradients > X in a specified region). The system needs to understand and support such queries. Note that queries such as these are NOT simply metadata queries. They will need to deal with subqueries that are processing-based (e.g., gradient may be computed rather than stored as a separate product). Such methods need to be supported WITHIN EOSDIS — otherwise, accessibility and sharing will be lost.

(Skole - UNH) Dave's model is that this "architecture" should hold through EOSDIS, with the project supplying core interoperability support, and the SCFs providing the information management layer. The belief is that EVERYTHING beyond level 1 and 2 processing would be done at the SCF, due to SCF-specific algorithm requirements and tracking capabilities.

(Skole - UNH) He is a strong proponent of science driving the systems. His example was that the LANDSAT Pathfinder was designed to answer the science question of "where is the missing carbon in the planetary carbon cycle associated with the process of deforestation (tropical rain forests are thought to be CO_2 sinks for the planet)?" That basic question drove the selection of a simple (UTM) coordinate system with spatial resolution specific to the problem. There is little agreement among the various scientists involved, as to what the best coordinate system and resolution is "in general".

3. Support an interactive investigation capability

This driver reflects the scientists' desires for a streamlined environment in which data is readily available for interactive investigations. "Interactions" can include simple display of images, more complex data manipulations and visualization, and algorithmic processing (e.g., to attempt to extract new features).

Science investigations are interactive by nature, with scientists posing questions, analyzing data to answer those questions, posing a new set of questions, etc. Current practice is such that a great deal of time is spent hassling with the mechanics of data conversion and formatting. By handling such things smoothly, ECS will support more efficient investigations, allowing the scientist to focus on the details of the research and not the mechanics of information retrieval and formatting. ECS needs to provide a seamless environment "from desktop to DAAC," in which SCFs and individual investigators can become an integral part of the core system. In addition to providing rapid access to data, the system must allow for the posing of "new" questions — e.g., questions about image characteristics that have never before been computed.

There are a few key implications of this driver:

» Data should be easily accessible; system management functionality should be unobtrusive

» Data format conversion should be handled by user-supplied methods and/or common data formats.

» The specification of data may be formulaic, and may include user-supplied computational "methods"

Substantiating comments

(Skole - UNH) The pathfinder approach is at the heart of the EOSDIS program — i.e., the use of rapid prototyping to develop experience with existing global datasets before platform launch, and hence develop "pathfinders".

(OSU) HAIS needs to take a look at a much more interactive investigation model. This model uses frequent querying of the data from the SCF, interspersed with computational activity / requests. The interaction is quite fine grained, inconsistent with a coarse product ordering / generation capability.

(OSU) The SRR focused on data volumes. An investigation of data rates was missing. There was no emphasis on SCF-DAAC interaction. There was little analysis of how significant changes in network bandwidth might affect the way scientists work (i.e., the paradigm shift). The desktop must be considered a PART of EOSDIS that evolves with (in fact pushes evolution of) the system. ...

(OSU) ECS must address the ease with which scientists can try out their ideas. For example, it shouldn't take 2-3 days for a scientist to obtain data, and write appropriate extraction/filter routines to get model and instrument data into a compatible format. The hidden cost to ECS is manpower! If that cost is simply pushed externally (i.e., to the scientists), there is the

possibility that the science will not get done. The ECS infrastructure must minimize the external science costs.

(OSU) The OSU guys urged a look at "translator objects" as an alternative to common data formats.

(Dozier) ... the system should be driven by the need to get bits to the users, rather than by the need to archive bits.

4. Support an information-rich data pyramid

One of the key characteristics the system should possess is the ability to retain data "lineage," or "heritage". Data lineage is the bulk of information that describes how a particular data object (it may be a product, derived metadata, etc.) was generated. Lineage would include original source information (platform, sensor, date, time), plus any additional information used in processing (e.g., specific versions of calibration files), including algorithms, additional input datasets, etc. A rich data lineage in the "data pyramid" supports fundamental research questions about data and its sources, and allows scientists to explore the effects of alternative models in their work.

Access to lineage information should extend into tools available to researchers. In particular, the ability to obtain information about a data product's lineage from within a visualization environment is desired. Visualization often includes separation of the actual data values from the visualized representation (due, e.g., to histogram stretching for visual effect). The ability to (recursively) probe into the origins of data items ("drill down") would support researchers in better understanding what they are viewing.

Substantiating comments

(Skole - UNH) Dave suggested the NASA/NOAA Pathfinder project as a good production model to analyze. AVHRR 1km data is obtained at a number of internationally distributed, cooperating locations throughout the world via HRPT stations. Gaps are filled using NOAA data, and tapes are provided to the LP DAAC. There, the orbits are stitched together, and correction/projection algorithms applied to generate the base "product". The system then maintains a lineage tracking system to track the processing history of each pixel in the product. A working group was able to achieve community consensus on algorithms, atmospheric corrections, etc. through a set of focused processing stream element "workshops". A summary is available in IGBP Report #20.

(Aber/Martin - UNH) Both these researchers seemed to think that it is important to retrieve the raw data that has gone into other peoples' published results. They would like to investigate the differences that different processing makes to the results. They said that different processing could (e.g., for atmospheric distortion) change estimate of the bottom line (e.g., forest canopy

coverage) by as much as 50%. This potential for large differences points out the need to carefully document the transformations that are made to the data.

(OSU) They raise as one of the issues with current visualization tools the problem of "heritage tracing". The tools provide outstanding presentation and manipulation facilities, but lose track of the scientific basis ("validity") for the data being visualized. For example, in viewing the sea surface temperature, they've lost the navigation and temporal information associated with the data. In fact, they've even lost the actual temperature information, as they perform histogram equalization and stretch analyses to optimize the visual presentation of information. They believe there is a need for a tool with stronger lineage tracing. [This appears to be a uniform visualization issue — what is termed "drilldown" capability in other application areas].

5. Support the integration of independent investigator tools

Rather than attempting to "reinvent the wheel" developing yet another visualization or analysis tool, ECS should work at providing the infrastructure and interoperability services to support researchers in using their own tools. This approach leverages the efforts that have gone into developing a rich set of existing investigative tools, and encourages future improvements in a competitive environment. This approach is key to encouraging scientific creativity in developing tools appropriate to the research task at hand.

While ECS may wish to provide some simple set of basic analysis and visualization tools, these should be viewed as an optional set that can be replaced by more powerful or science specific tools. Where possible, ECS should be attempting to develop capabilities in new areas, rather than attempting to reproduce capabilities that already exist in commercial products. A "workbench" with some key rudimentary tools, and into which investigators can add their own tools, appears to be an appropriate metaphor.

Substantiating comments

(Moore) The tools that the project puts in the hand of investigators are key to the success of EOSDIS. Providing, e.g., an alternative visualization tool that has one real neat feature, but is lacking in other dimensions compared with existing tools is a recipe for failure.

Tools themes:

- The research environments of the SCFs are heterogeneous
- The tools being used are NOT the same throughout the community
- The project should NOT be developing a single monolithic toolkit

- Rather, it should be building a common core around interprocess communications (DCE) and file transfer to support existing and future environments

(Walstad - OSU) In ocean dynamics models, data volume is the big problem. The key to dealing with these large volumes of data is the availability of flexible algorithms and processing tools. If the data or tools cannot be made readily available to the researcher in the form they want to use, the science often doesn't get done. Far too much time is spent programming. The use of common data formats for different data sets will be crucial. Translation of data into common formats should be the role of a data center(s); data analysis should be the role of the researcher.

(OSU) They use an extensive array of tools to support visualization and animation. These are clearly key components in the SCF of the near future (and the present!).

6. Support user-to-user collaboration

The system should facilitate collaboration among scientists, especially "early" in the investigation of new data sources and algorithms (i.e., products). There needs to be an easy way for researchers to share information, and this sharing should be supported by the system. The concept described here is one in which scientists could insert ("publish") new data and algorithms into the system, appropriately characterized, for investigation and use by colleagues. They should then be able to provide shorthand "references" to the data and algorithms and pass those to collaborators for their use.

This collaboration should be able to occur "point-to-point" — e.g., directly between investigators, and without significant system overhead or intervention.

Substantiating comments

(Dozier) There is a lot happening "outside the DAACs." Point-to-point external interactions must be folded-in.

(Skole - UNH) Dave's "model" is consistent with a more distributed "plug and play" service architecture, in which contributing researchers "publish" their data and access methods for the rest of the world. These products, however, are maintained (and even generated) locally, in response to dynamic requests from the community. We talked a bit about such a service-oriented architecture, including the need for local control of security and resource consumption. One of the concepts we discussed is the idea of SCFs acting as mini-PGS's, both ingesting and distributing data (to other SCFs).

7. Provide distributed administration and control

The centralized SMC functionality and authority presented at SRR needs to be replaced by a decentralized, more autonomous approach. Granting of access privileges, especially to local facilities that may be part of EOSDIS (DAACs or SCFs), should be done at the local level. A single, centralized access authority, or one relying on investigator's home institutions is not acceptable. Additionally, there should be some amount of system "access" even for "unsponsored" (new) users.

Substantiating comments

(Emery) The system architecture presented at SRR was a centralized architecture that will not work. Concerns include:

- the reliance on sponsoring "institutions". The system needs to deal with researchers and research groups as individuals. Many universities will not have appropriate infrastructure to act as sponsoring institutions.
- the SMC is a potential system bottleneck and single point of failure. Access to otherwise available data might be prevented by a problem within the SMC (e.g., maintenance). Distribute and replicate the SMC functionality to prevent this.
- additionally, there is concern that the SMC is too centralized, and will be a single point of failure that could prevent access by new or existing users to part or all of the system. The SMC functionality should be logically and physically distributed among the physically distributed system components.

(Dozier) The problem of dispersing responsibility, authority, and resources is tied to the development of a distributed system, and must be addressed as well.

2.3. Technology drivers

This subsection focuses on drivers that are associated with key technology advances in hardware, software, networking, and "horizontal" applications. These technology advances will define the computing environment of the future, and hence will greatly impact the way in which the scientists will employ the resources of ECS in their research efforts.

1. Software

The advances in information technology software is proliferating at a rapid pace. The program needs to be more visionary in identifying key information technologies and in infusing them into the system design early. Among key technologies are:

- Distributed processing and interoperability protocols

  This includes, in addition to OSF/DCE, inclusion of higher level communications and interoperability protocols such as Ellery Open Systems, CORBA, etc.
- Object-oriented Databases and extended relational models

The need for extensions to traditional relational models seems clear. Investigation of object oriented database technologies should be an important part of ECS.

- Operating systems

  This includes pledging support for a heterogeneous desktop environment that is likely to include Unix, Windows NT, Apple's new operating system, as well as others that are likely to develop.

Substantiating comments

(Dozier) The different approaches among different elements of EOSDIS has been lost. This goes beyond "DAAC unique functionality" to the concept of supporting fundamentally different operational infrastructures, including tools, databases, etc. The result of a homogenous system will be guaranteed mediocrity, believes Jeff, as individual creativity will be squelched.

(Dozier) In particular, there is concern that version 3 (e.g.) will simply be a beefed up version of v. 1. The program appears to have lost touch with fundamental technology and scientific drivers that would cause a v.3 system to look quite different from a v.1 system.

(Dozier) There have been significant technological drivers and innovations since 1990 that should have been incorporated in the current version of the architecture. He is very surprised (dismayed) that these innovations weren't implemented in the post-Version 0 efforts.

(Abbott) Abbott stressed their role as leading edge technology consumers. The school was a beta site for NT, and they are currently in the process of re-wiring the building to bring fiber into every office for digital video capability. Abbott is very keen on insuring that lots of bandwidth and adequate cycles are available on everyone's desktop because "that's where the action is."

(OSU) OSU has been involved in Beta testing Windows NT, and is using PCs and Macs increasingly to perform tasks traditionally run on Unix workstations (e.g., they use Adobe Photoshop on a Mac @ $300, as opposed to the SGI version for $2000). The project needs to track changes in the desktop market that could significantly affect the characteristics of SCF and user workstations. These trends include Windows NT, the PowerPC, and the porting of Unix software to these platforms. Abbott mentioned the availability of a Beta version of IDL for the Macintosh.

(OSU) ... The desktop must be considered a PART of EOSDIS that evolves with (in fact pushes evolution of) the system. ECS should be tracking OS

futures (Sun/Spring, Microsoft NT/Cairo, etc.) that will significantly change the way we work (e.g., desktop multimedia, etc.).

2. Networking (Gigabit links)

This is the "bandwidth is free" argument voiced at SRR. The point here is that substantial increases in network bandwidth can significantly affect the way you move data around the system, and the distribution and placement of various services. The system should be able to take advantage of advancing network technologies to change the way people will work (i.e., provide new services), in addition to simply providing more data faster. Higher speed networks will support a more seamless environment between users and DAACs. The system architecture should support high speed networks in enabling this seamless environment.

Additionally, there needs to be provisions for growth to support international data providers and consumers. If ECS expects to get data from international contributors, it must provide sufficient network bandwidth to enable their analysis and processing as well.

Substantiating comments

This driver was derived largely from the discussions at SRR, with the following additional comments:

(OSU) The SRR focused on data volumes. An investigation of data rates was missing. There was no emphasis on SCF-DAAC interaction. There was little analysis of how significant changes in network bandwidth might affect the way scientists work (i.e., the paradigm shift). The desktop must be considered a PART of EOSDIS that evolves with (in fact pushes evolution of) the system. ...

(Diogenes - UNH) There is great concern about the support for international partners. Networking infrastructure support seems to be a big problem currently — current EOSDIS plans don't appear to address networking issues except through government furnished internet support. In order to support international researchers (and pull them in as data providers as well), we will need to address this issue.

3. Processing (MPP proliferation, high performance workstations)

This is the "flops are free" argument voiced at SRR. And while flops may not literally be free, the continual improvement in hardware price/performance needs to be considered in the architectural approach to ECS. In particular, ECS should be able to take advantage of the substantial computing power that exist in SCFs via appropriate distributed computing strategies. Additionally, the use of Massively Parallel Processors (MPPs) should be considered where appropriate to leverage contract hardware dollars for "tall pole" algorithms that can benefit from a data parallel programming approach.

Substantiating comments

This driver was derived primarily from the discussions at SRR.

4. Multimedia (collaboration environments, videoconferencing)

ECS needs to address support for multimedia technologies such as collaboration environments, videoconferencing, etc. These technologies will be an integral part of the scientific investigator's toolbox, and need to be supported within ECS. Collaboration environments will allow scientists to share data and convene "meetings" to discuss important new findings. Digital animation and compositing will be fundamental data preparation and presentation tools. Videoconferencing will be a common mechanism for putting together impromptu meetings to discuss relevant topics of research. Video and voice annotations will be used throughout the system, both at data capture (e.g., recorded field notes), and as supplementary data annotations.

Substantiating comments

(Skole - UNH)	They are working on a "hyperGIS," that will include multimedia based "ground truth" (audio, video recordings, pictures, etc.), notes, and articles in a single DB. They are putting components of this together using tools in the SunOS.
(OSU)	They use an extensive array of tools to support visualization and animation. These are clearly key components in the SCF of the near future (and the present!).

3. Mandate

Based on SRR feedback, and subsequent interaction with the science community (as described herein), we have been re-evaluating some of the key concepts in the baseline system architecture. This section presents what we are taking to be a "mandate" from the scientific community to develop a new conceptual architecture capable of meeting the needs expressed in this document. We have derived this mandate from a combination of the drivers presented above and additional analyses in response to science community and NASA direction. Together, the elements of the mandate comprise the building blocks for an evolvable system.

That mandate is summarized in the following set of system design guidelines.

	Move from ...		*Towards ...*
Ø	"Element"-oriented architecture	√	Service-oriented architecture
Ø	Replicated centralized approach	√	Logically distributed approach
Ø	Metadata / data distinction	√	"Data is data"
Ø	DAAC-centric implementation	√	Extended provider implementation
Ø	Across-DAAC homogeneity	√	DAAC autonomy / heterogeneity
Ø	Centralized administration	√	Distributed control / authority
Ø	Product approval and ordering	√	Product "publishing" and "access"

The set of mandates is recreated below, with textual descriptions interspersed between each of the individual guidelines.

	Move from ...		*Towards ...*
Ø	"Element"-oriented architecture	√	Service-oriented architecture

The baseline architecture presented at SRR included architectural biases introduced by the element level functional decomposition. These biases included architectural artifacts that should not be introduced as requirements — for example, the artificial delineations between "data" and "metadata". Migration towards a service-oriented architecture in which services can be replicated and distributed across physical components will provide a better conceptual architecture on which to build.

Ø	Replicated centralized approach	√	Logically distributed approach

The baseline architecture instituted a "replicated centralized" approach to resource and information distribution. The IMS-PGS-DADS element architecture was simply replicated across physically distributed locations, with all sites having to provide equal and full

functionality. A logically distributed approach, in which services are available regardless of their physical location, will provide ECS with a more flexible foundation upon which to evolve.

Ø Metadata / data distinction √ "Data is data"

The distinction between data and metadata resulting from an element-wise system partitioning seems overly restrictive. The SRR presentation seemed to imply that metadata included only attribute-value descriptors that have been pre-computed on the data (i.e., data "indices"), and that the various products constituted "data". The system should support a "data is data" concept in which an entire range of data derivations are provided, some at time of ingest (computed or user/source provided), and some later, perhaps even generated "on demand' in response to a specific query. This unifying concept will support extensibility of the information management "database" to provide the necessary system evolvability as knowledge continually increases about data objects in the system.

Ø DAAC-centric implementation √ Extended provider implementation

The baseline architecture appeared to make no provisions for data service providers other than DAACs. Because of the dynamic nature of many of the research data products, it will be necessary to readily support the generation and distribution of products created at other sites, in particular, at the SCFs. The dynamic product lifecycle and support for additional data providers may also require a re-analysis of the balance between process-on-demand and routine production data products. The extended provider model should also provide a more flexible architectural foundation on which to evolve the ECS.

Ø Across-DAAC homogeneity √ DAAC autonomy / heterogeneity

The baseline architecture included a homogeneous model of system components from DAAC to DAAC (for example, employing the same IMS database and metadata schema across all DAACs). The concept of DAAC unique capabilities to provide value added services appeared either not to be supported, or to be supported in a restrictive fashion. DAAC autonomy and heterogeneity should be the cornerstone of the system design, rather than an anomaly.

Ø Centralized administration √ Distributed control / authority

The baseline architecture was based on centralized administration (through a single, centralized SMC), and a network of sponsoring institutions. This organization is unacceptable to a widely distributed body of researchers who can't necessarily rely on their sponsoring organizations for support. A more distributed control structure with appropriately distributed resource and access control is required. Such a structure will eliminate the single point of failure SMC, and will support evolution of value added provider services in a "free market" type of system.

Ø Product approval and ordering √ Product "publishing" and "access"

The baseline system included a perceived "heavyweight" product approval process through IWG algorithm review. Additionally, the concept of product "ordering" is a more heavyweight concept than simple data "access". The system should include provisions for data provider "publishing" in more of a free market approach, and should support unobtrusive direct access to data once found through querying and search mechanisms. The concept of

"mounting" and "using" remote data objects (e.g., through universal object handles) should be supported.

An approach employing these design guidelines is in fact, consistent with current information technology trends towards distributed, service-oriented information architectures. Such an approach will support system evolution both in scale and in function, enabling ECS to develop as an information system that will promote advances in earth science research.

4. Processes and relationships

An important goal of the ECS Program is to provide a highly adaptable system that is responsive to the evolving needs of the Earth science community. This must be accomplished in close cooperation with the users of the system and with the full realization that an adaptable ECS requires the continual infusion of advanced information science technology. The ECS Program Office is doing the following to facilitate communication with the science community and to ensure that the science users effectively participate in the development of the ECS system.

- Develop the Science Office as a program driver for system requirements and a sounding board for design tradeoffs and overall guidance.
- Conduct frequent visits to user research facilities to obtain first-hand knowledge of research issues driving specific user requirements which, in turn, will be used to assist in future program development.
- Organize and implement science and technology workshops which will provide a forum for discussion of Earth science-computer science research issues, user needs, and requirements for an evolving ECS.
- Establish a Computer Science Advisory Panel comprising internationally recognized experts in computer and information sciences which will provide direct input concerning program direction and introduction of state-of-the-art technology and concepts into the ECS Program.
- Actively participate in advanced technology programs and consortia, and where appropriate transfer technology and knowledge gained from those activities into the ECS.
- Develop a more comprehensive collaborative prototyping and development program to leverage the significant efforts in the Earth science and Computer science communities. These activities should include studying and building upon V0 DAAC activities, national infrastructure program activities, and key information technology research activities in academia and industry.
- Administer ECS Program-funded alternative architecture studies independently conducted by external educational institutions and research consortia.
- Organize and manage Program-funded university research projects on selected computer and information science topics relevant to ECS development and improvement.
- Develop and implement appropriate user feedback mechanisms to effectively elicit opinions and recommendations from the science users concerning specific aspects of ECS operation and performance.

Acceptance and endorsement of the ECS by the science community are the touchstones of the program. The Program Office fully recognizes that the Earth Observing System is a science-driven program and that development of an ECS which addresses the needs of the science users is the principal program imperative. By instituting and sponsoring the activities listed above, the ECS Program Office hopes to establish a close working relationship with the computer and Earth science

communities that is essential to effectively define program directions in an environment of evolving user needs and continually improving technology.